
Enhanced Sampling Toolkit Documentation

Release 1.0

Jeremy Tempkin

April 07, 2015

CONTENTS

1 Walker API	3
2 LAMMPS Walker Module	5
3 Indices and tables	7
Python Module Index	9
Index	11

Contents:

WALKER API

NOTE: we may want to use abstract properties in the future.

class `walker.velocityWalker`

This is an abstract class for dynamics which have a velocity component, such as a protein under Langevin dynamics. It extends the walker abstract class by adding additional abstract methods that are necessary for a walker that has a velocity.

drawVel ()

Draws a new value of the velocities for the walker.

reverseVel ()

This function reverses the velocity of the walker.

class `walker.walker`

Defines an abstract walker object. It functions as an interface, providing methods that other walkers should implement.

close ()

Destroys the walker.

equilibrate (*colvar*)

This function sets the walker at a specific configuration in the collective variable space. At minimum, any implementation will need to give it the configuration.

getColvars ()

This function returns the location of the walker in the collective variable space.

getConfig ()

This function should return the configuration of the walker in configuration space.

propagate (*numsteps*)

This function propagates the simulation forward a given number of steps. It takes as an argument at least *numsteps*, the number of steps to propagate forward.

setConfig (*configuration*)

This function should set the system at a specific place in configuration space. At minimum, it should take some sort of specification of the configuration.

LAMMPS WALKER MODULE

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*

W

walker, 3

C

`close()` (walker.walker method), 3

D

`drawVel()` (walker.velocityWalker method), 3

E

`equilibrate()` (walker.walker method), 3

G

`getColvars()` (walker.walker method), 3

`getConfig()` (walker.walker method), 3

P

`propagate()` (walker.walker method), 3

R

`reverseVel()` (walker.velocityWalker method), 3

S

`setConfig()` (walker.walker method), 3

V

`velocityWalker` (class in walker), 3

W

`walker` (class in walker), 3

`walker` (module), 3